



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/691,886

10/23/2003

Timothy P. McKee

MFCP.108799

8989

45809

7590

07/11/2008

SHOOK, HARDY & BACON L.L.P.  
(c/o MICROSOFT CORPORATION)  
INTELLECTUAL PROPERTY DEPARTMENT  
2555 GRAND BOULEVARD  
KANSAS CITY, MO 64108-2613

EXAMINER

STACE, BRENT S

ART UNIT

PAPER NUMBER

2161

MAIL DATE

DELIVERY MODE

07/11/2008

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b> 10/691,886	<b>Applicant(s)</b> MCKEE ET AL.	
	<b>Examiner</b> BRENT STACE	<b>Art Unit</b> 2161	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 14 April 2008.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-41 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-41 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 10 October 2007 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                     | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____                                      |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____  | 6) <input type="checkbox"/> Other: _____                          |

## **DETAILED ACTION**

### ***Remarks***

1. This communication is responsive to the amendment filed April 14<sup>th</sup>, 2008. Claims 1-41 are pending. In the amendment filed April 14<sup>th</sup>, 2008, no Claims are amended, and Claims 1, 30, and 41 are independent. The examiner acknowledges that no new matter was introduced and the claims are supported by the specification. This action is made FINAL.

### ***Response to Arguments***

2. Some of the Applicant's arguments filed April 14<sup>th</sup>, 2008 with respect to claims 1-41 have been considered but are not persuasive.

3. As to the applicant's arguments with respect to Claims 1-41 for the prior art(s) allegedly not teaching "at least a portion of the items contain metadata defining a visual representation of at least one of said plurality of items and defining a plurality of user-selectable tasks," the examiner respectfully disagrees.

Toolbox, p.815, paragraph under "Creating Customized Document Icons" to the second to last paragraph with Stuffit, p. 84, last two full paragraphs relating to StuffitCM with Stuffit, p. 86, the figure was used to reject this claimed limitation. The citings in Toolbox teach that customized icons (visual representations for items) can be created for documents in the Mac OS. These custom icons are stored in the resource fork of a file (metadata contained in an item). The citings in Stuffit teach that when a user

control-clicks on a file/folder/archive a contextual menu will appear that only displays "options that relate to the file or folder selected." As such, the Finder/Mac OS must know what it is the user is clicking on in order to determine what kind of options to show in the contextual menu. p.816 of Toolbox, paragraphs under "Creating File Reference Resources" shows that a file type is associated with each file's resource fork. Since Stuffit teaches that it can differentiate between files, folder and archives, stuffit has the ability to access metadata contained in the item and use that to define/obtain user-selectable tasks in the contextual menu. The indication of user-selectable tasks is displayed to the user via the contextual menu items selections. The contextual menu item has tasks in it that are user-selectable and will launch an application in response to a user input selecting one or more tasks (stuffit and mail will open and email application). When the user selects stuff and mail, the user-selectable task will be performed with respect to the requested item.

Additionally, MacOSXHints, pages 1-2, paragraphs 3, 6, and 8 (using logic from Landau, p. 4, everything below Fig. 6.43) was used to reject this claimed limitation. With Landau, the normally invisible .DS\_Store files from MacOSXHints are made visible. These .DS\_Store files contain metadata defining how items (folders and files) appear on the screen within a window (order, location, etc.). As such, there is at least a portion of the items (the .DS\_Store file) containing metadata that defines a visual representation (layout of items) of at least one of said plurality of items. This .DS\_Store file also defines a plurality of user-selectable tasks in that the .DS\_Store file is accessed to determine how to draw the window (position, size, list/icons/column view options,

"keep arranged by" options). All these parameters are user-selectable tasks, tasks done by drawing the window. By modifying window parameters in the finder or changing the view options in the Finder (e.g. via command-j), the user creates different user-selectable tasks. The changing of view options via command-j is an "indication of said plurality of user-selectable tasks" where the Finder's application for applying a view option is launched when the user changes the options.

4. In response to applicant's argument that there is no suggestion to combine Toolbox and Stuffit, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, suggestion to combine the references is found in the references themselves. Both Toolbox and Stuffit are directed towards using a GUI on a computer with icons for files, folders, and applications. Stuffit's invention would have been expected to successfully work well with Toolbox's invention because both inventions use the Mac OS GUI with icons for files, folders, and applications. It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Stuffit and Toolbox before him/her to take the contextual menus from Stuffit and install it into the invention of Toolbox, thereby offering the obvious advantage of sending email with compressed attachments all with point and click simplicity (Stuffit, p. 3, lines 1-2).

Additionally, the applicant's argument appears to be invalid since the argument relies on "Toolbox's .DS\_Store files." The argument appears to be invalid because Toolbox doesn't teach anything about .DS\_Store files, MacOSXHints teaches the .DS\_Store files.

5. The other claims argued merely because of a dependency on a previously argued claim(s) in the arguments presented to the examiner, filed April 14<sup>th</sup>, 2008, are moot in view of the examiner's interpretation of the claims and art and are still considered rejected based on their respective rejections from at least a prior Office action (part(s) of recited again below).

### ***Response to Amendment***

#### ***Prior Art***

6. The examiner would like to note that the prior art of record and used in the rejections below, MacOSXHints (<http://www.macosxhints.com/article.php?story=20030305025744788>) and MacOSXHints2 (<http://www.macosxhints.com/article.php?story=20030409015020645>), were published less than a year prior to the filing date of the present application making them prior art under 35 USC § 102(a). However, these references refer solely to an operating system's behavior where the operating system's public release date is a 35 USC § 102(b) date.

### ***Specification***

7. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

### ***Drawings***

8. Since the lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors, Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the drawings. For example, the drawings should be carefully checked to ensure that all reference numerals are described in the specification, that no one reference numeral describes two separate drawing elements, or that the specification contains no reference to numerals not in the drawings.

### ***Claim Rejections - 35 USC § 103***

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein

were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

11. Claims 1-41 are rejected under 35 U.S.C. 103(a) as being unpatentable over “Inside Macintosh: Macintosh Toolbox Essentials” (Toolbox) in view of StuffIt Deluxe User’s Guide (StuffIt).

For **Claim 1**, Toolbox teaches: “A computer system for presenting a plurality of items of multiple types to a user, [Toolbox, p.802, Fig. 7-1 with the Toolbox, p.801, last paragraph] the system comprising:

- a universal data store containing said plurality of items stored in accordance with a universal data schema, [Toolbox, p.802, Fig. 7-1 with Toolbox, p. 803, paragraph under Fig. 7-2] wherein at least a portion of the items contain metadata defining a visual representation of at least one of said plurality of items... [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph] and
- a shell for presenting said plurality of items in the universal data store to a user, [Toolbox, p.815, paragraph under “Creating Customized Document Icons” with Toolbox, p.802, Fig. 7-1] wherein the shell is configured to present an item containing said metadata to the user with said visual representation...defined by



said metadata contained in said item [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph] and further configured to present the items without said corresponding metadata to the user in accordance with a default display view” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

Toolbox discloses the above limitations but does not expressly teach:

- “...and defining a plurality of user-selectable tasks;
- ...and with an indication of said plurality of user-selectable tasks...wherein said shell is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks.”

With respect to Claim 1, an analogous art, Stufflt, teaches:

- “...and defining a plurality of user-selectable tasks; [Stufflt, p. 84, last two full paragraphs relating to StuffltCM with Stufflt, p. 86, the figure]
- ...and with an indication of said plurality of user-selectable tasks [Stufflt, p. 84, last two full paragraphs relating to StuffltCM with Stufflt, p. 86, the figure]...wherein said shell is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks” [Stufflt, p. 84, last two full paragraphs relating to StuffltCM with Stufflt, p. 86, the figure with Stufflt, p. 78, paragraph above number 3 in “Stuff and Mail...”].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Stufflt and Toolbox before him/her to combine Stufflt

with Toolbox because both inventions are directed towards using a GUI on a computer with icons for files, folders, and applications.

Stufflt's invention would have been expected to successfully work well with Toolbox's invention because both inventions use the Mac OS GUI with icons for files, folders, and applications. Toolbox discloses a how icons work in the Mac OS comprising default icons, custom icons, and icons in the file's bundle resource. However, Toolbox does not expressly disclose defining, indicating or launching application(s) of user-selectable tasks. Stufflt discloses a compression utility with Mac OS integration comprising contextual menus that can stuff and email archived file system objects.

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Stufflt and Toolbox before him/her to take the contextual menus from Stufflt and install it into the invention of Toolbox, thereby offering the obvious advantage of sending email with compressed attachments all with point and click simplicity (Stufflt, p. 3, lines 1-2).

**Claim 2** can be mapped to Toolbox (as modified by Stufflt) as follows: "The computer system of Claim 1, wherein said universal data schema includes one or more declared properties" [Toolbox, p.815, third to last paragraph, note the Finder flags, p. 7-34 says that finder flags are per document and with the document].

**Claim 3** can be mapped to Toolbox (as modified by Stufflt) as follows: "The computer system of Claim 2, wherein each of said plurality of items includes

declarations associated with said one or more declared properties” [Toolbox, p.815, third to last paragraph].

**Claim 4** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 2, wherein said shell utilizes said one or more declared properties to present the items without said corresponding metadata in accordance with said default display view” [Toolbox, p.815, paragraph under “Creating Customized Document Icons”].

**Claim 5** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 1, wherein said metadata includes item decoration data” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 6** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 5, wherein said item decoration data identifies a subset of said metadata containing high value data” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 7** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 6, wherein said high value data is associated with information to be displayed to a user” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 8** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 5, wherein said item decoration data includes data indicating a format aspect associated with presentment of said metadata to a user” [Toolbox,

p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 9** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 8, wherein said format aspect is selected from the group consisting of data formatting, sort order, icon formatting, and associated controls” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 10** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 8, wherein said format aspect includes one or more verbs, said verbs being associated with operations to be performed on selected items” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph with Toolbox, p.802, Fig. 7-1 or Stufflt, p. 86, the figure].

**Claim 11** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 10, wherein said operations are performed by one or more applications launched to perform said verbs with respect to said selected items” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph with Toolbox, p.802, Fig. 7-1 Stufflt, p. 86, the figure].

**Claim 12** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 1, wherein said shell is further configured to present a set of items selected from said plurality of items according to a shell decoration view” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 13** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 12, wherein said set of items includes one or more items having metadata stored in accordance with a first item decoration schema and further includes one or more items having metadata stored in accordance with a second item decoration schema” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 14** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 12, wherein said shell decoration view includes one or more display aspects appropriate for displaying each of said set of items to a user” [Toolbox, p. 848, frView with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 15** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 14, wherein each of said set of items include a common characteristic” [Toolbox, p.802, Fig. 7-1 with the Toolbox, p.801, last paragraph].

**Claim 16** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 15, wherein said common characteristic is selected from the group consisting of document file, image file, audio file, and video file” [Toolbox, p.802, Fig. 7-1 with the Toolbox, p.801, last paragraph].

**Claim 17** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 12, wherein said one or more of said set of items are associated with an item decoration view that conflicts with said shell decoration view” [Toolbox, p.855, last paragraph].

**Claim 18** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 12, wherein said shell is further configured to present one or more of said set of items with decorative elements associated with an item decoration view and wherein said decorative elements do not conflict with said shell decoration view” [Toolbox, p.855, last paragraph with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 19** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 1, wherein said shell is further configured to present items selected from said plurality of items according to an explorer display view” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 20** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more decorative properties” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 21** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more shell display views, and wherein one of said shell display views is utilized to display a set of explorer items” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 22** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 21, wherein said shell display view includes one or more display aspects appropriate for displaying each of said set of explorer items to a user” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 23** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 22, wherein each of said set of explorer items include a common characteristic” [Toolbox, p.802, Fig. 7-1 with Toolbox, p. 848, frView].

**Claim 24** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 21, wherein said explorer display view is configured to utilize a shell display view not included with an explorer to display an item which cannot be displayed according to said one or more shell display views included with said explorer display view” [Toolbox, p. 855, last full paragraph].

**Claim 25** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 21, wherein said explorer display view is configured to display an explorer item which is not associated with a shell display view according to an item decoration view” [Toolbox, p. 855, last full paragraph with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 26** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 21, wherein said explorer display view is configured to display an explorer item which is not associated with an item decoration view according

to said default display view” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 27** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more data queries associated with said explorer display view” [Toolbox, p. 855, last full paragraph].

**Claim 28** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more verbs, said verbs being associated with an operation to be performed on a selected explorer item” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph with Toolbox, p.802, Fig. 7-1 or Stufflt, p. 86, the figure].

**Claim 29** can be mapped to Toolbox (as modified by Stufflt) as follows: “The computer system of Claim 28, wherein said operation is performed by an application launched to perform at least one of said one or more verbs with respect to said selected explorer item” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph with Toolbox, p.802, Fig. 7-1 or Stufflt, p. 86, the figure].

For **Claim 30**, Toolbox teaches: “A computer implemented method for presenting a plurality of items stored in a universal data store to a user, [Toolbox, p.802, Fig. 7-1 with the Toolbox, p.801, last paragraph with Toolbox, p.802, Fig. 7-1] the method comprising:



- accessing said universal data store in response to a request to present one or more of said plurality of items to the user, [Toolbox, p.802, Fig. 7-1 with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph] wherein said plurality of items are stored are in accordance with a universal data schema, [Toolbox, p. 803, paragraph under Fig. 7-2] and wherein at least a portion of said plurality of items contain metadata stored in accordance with an item decoration schema, wherein said metadata defines a visual representation of an item [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph]...
- for at least a portion of said one or more requested items, accessing said metadata contained in a requested item to define said visual representation of said requested item [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph]...
- presenting said one or more requested items to the user, [Toolbox, p.815, paragraph under “Creating Customized Document Icons” with Toolbox, p.802, Fig. 7-1] with said visual representation and [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph]; [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

Toolbox discloses the above limitations but does not expressly teach:

- ...and further defines a plurality of user-selectable tasks that may be performed with respect to said item;

Art Unit: 2161

- ...receiving a user input selecting one of a plurality of user-selectable tasks;
- ...directing said at least one application to perform the selected user-selectable task with respect to said one or more requested items”
- “...and to obtain said plurality of user-selectable tasks stored in said metadata;
- ...with an indication of said plurality of user-selectable tasks defined by said metadata
- ...launching at least one application in response to said user input.”

With respect to Claim 30, an analogous art, StuffIt, teaches:

- ...and further defines a plurality of user-selectable tasks that may be performed with respect to said item; [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]
- ...receiving a user input selecting one of a plurality of user-selectable tasks; [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]
- directing said at least one application to perform the selected user-selectable task with respect to said one or more requested items” [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure with StuffIt, p. 78, paragraph above number 3 in “Stuff and Mail...”]
- “...and to obtain said plurality of user-selectable tasks stored in said metadata; [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]

- ...with an indication of said plurality of user-selectable tasks defined by said metadata [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]
- ... launching at least one application in response to said user input” [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure with StuffIt, p. 78, paragraph above number 3 in “Stuff and Mail...”].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of StuffIt and Toolbox before him/her to combine StuffIt with Toolbox because both inventions are directed towards using a GUI on a computer with icons for files, folders, and applications.

StuffIt's invention would have been expected to successfully work well with Toolbox's invention because both inventions use the Mac OS GUI with icons for files, folders, and applications. Toolbox discloses a how icons work in the Mac OS comprising default icons, custom icons, and icons in the file's bundle resource. However, Toolbox does not expressly disclose defining, indicating or launching application(s) of user-selectable tasks. StuffIt discloses a compression utility with Mac OS integration comprising contextual menus that can stuff and email archived file system objects.

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of StuffIt and Toolbox before him/her to take the contextual menus from StuffIt and install it into the invention of Toolbox, thereby offering

the obvious advantage of sending email with compressed attachments all with point and click simplicity (Stufflt, p. 3, lines 1-2).

**Claim 31** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 30, wherein said metadata includes data indicating one or more decorative aspects associated with presentment of said metadata to a user” [Toolbox, p.815, paragraph under “Creating Customized Document Icons”].

**Claim 32** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 31, wherein presenting a requested item with said corresponding metadata includes presenting said metadata according to one or more of said decorative aspects” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 33** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 31, wherein said method further comprises presenting a set of items according to a shell decoration view, [Toolbox, p. 848, frView] wherein said set of items includes one or more items having metadata stored in accordance with a first item decoration schema and further includes one or more items having metadata stored in accordance with a second item decoration schema” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 34** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 33, wherein said shell decoration view includes one or more display aspects appropriate for displaying each of said set of items to a user” [Toolbox, p. 848,

frView with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 35** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 33, wherein presenting said set of items according to a shell decoration view includes presenting items having item decoration schema which conflicts with said shell decoration view according to said shell decoration view” [Toolbox, p.855, last paragraph].

**Claim 36** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 30, wherein said method further comprises presenting items according to an explorer display view” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 37** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 36, wherein said explorer display view includes one or more decorative properties [] and one or more shell display views, [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph] and wherein one or more of said shell display views are utilized to display a set of explorer items” [Toolbox, p. 848, frView with Toolbox, p. 855, last full paragraph with Toolbox, pgs. 857-858, last paragraph spanning the pages with the following paragraph].

**Claim 38** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 37, wherein an item which cannot be displayed according to said one

or more shell display views included with said explorer display view is presented according to a shell display view not included with said explorer display view” [Toolbox, p. 855, last full paragraph].

**Claim 39** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 37, wherein an item which cannot be displayed according to a shell display view is presented according to an item decoration view” [Toolbox, p. 855, last full paragraph with Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

**Claim 40** can be mapped to Toolbox (as modified by Stufflt) as follows: “The method of Claim 36, wherein said explorer display view includes one or more verbs, said verbs being associated with an operation to be performed on a selected explorer item” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph with Toolbox, p.802, Fig. 7-1 or Stufflt, p. 86, the figure].

For **Claim 41**, Toolbox teaches: “A shell embodied on one or more computer-readable media and executing on a computer for presenting a plurality of items stored in a universal data store to a user, [Toolbox, p.815, paragraph under “Creating Customized Document Icons” with Toolbox, p.802, Fig. 7-1] the shell comprising:

- a data store interaction component which retrieves metadata contained within one or more of said plurality of items in response to a request to present the one or more items to the user, [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph] wherein said plurality of items are stored in accordance with a universal data schema,

[Toolbox, p. 803, paragraph under Fig. 7-2] and wherein at least a portion of said plurality of items contain said metadata defining a visual representation of at least one of said plurality of items[Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph]...and

- a display presentation component which utilizes said retrieved metadata to present a display view of at least a portion of said one or more requested items, [Toolbox, p.815, paragraph under “Creating Customized Document Icons” with Toolbox, p.802, Fig. 7-1] wherein said display presentation component is configured to present a requested item containing said metadata to the user with said visual representation...defined by said metadata contained in said requested item [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph] and further configured to present a requested item without said corresponding metadata in accordance with a default display view...” [Toolbox, p.815, paragraph under “Creating Customized Document Icons” to the second to last paragraph].

Toolbox discloses the above limitations but does not expressly teach:

- “...and defining a plurality of user-selectable tasks;
- ...and with an indication of said plurality of user-selectable tasks
- ...wherein said display presentation component is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks.”

With respect to Claim 41, an analogous art, Stufflt, teaches:

- “...and defining a plurality of user-selectable tasks; [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]
- ...and with an indication of said plurality of user-selectable tasks [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure]
- ...wherein said display presentation component is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks” [StuffIt, p. 84, last two full paragraphs relating to StuffItCM with StuffIt, p. 86, the figure with StuffIt, p. 78, paragraph above number 3 in “Stuff and Mail...”].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of StuffIt and Toolbox before him/her to combine StuffIt with Toolbox because both inventions are directed towards using a GUI on a computer with icons for files, folders, and applications.

StuffIt’s invention would have been expected to successfully work well with Toolbox’s invention because both inventions use the Mac OS GUI with icons for files, folders, and applications. Toolbox discloses a how icons work in the Mac OS comprising default icons, custom icons, and icons in the file’s bundle resource. However, Toolbox does not expressly disclose defining, indicating or launching application(s) of user-selectable tasks. StuffIt discloses a compression utility with Mac OS integration comprising contextual menus that can stuff and email archived file system objects.



It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of StuffIt and Toolbox before him/her to take the contextual menus from StuffIt and install it into the invention of Toolbox, thereby offering the obvious advantage of sending email with compressed attachments all with point and click simplicity (StuffIt, p. 3, lines 1-2).

12. Claims 1-18, 30-35, and 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacOSXHints (<http://www.macosxhints.com/article.php?story=20030305025744788>) in view of "Mac OS X Disaster Relief: Troubleshooting Techniques to Help Fix It Yourself" (Landau).

For **Claim 1**, MacOSXHints teaches: "A computer system for presenting a plurality of items of multiple types to a user, [MacOSXHints, page 1, paragraph 3] the system comprising:

- a universal data store containing said plurality of items stored in accordance with a universal data schema, [MacOSXHints, page 1, paragraph 3 with MacOSXHints, page 2, paragraphs 6 and 8] wherein at least a portion of the items contain metadata defining a visual representation of at least one of said plurality of items and defining a plurality of user-selectable tasks; [MacOSXHints, pages 1-2, paragraphs 3, 6, and 8] and
- a shell for presenting said plurality of items in the universal data store to a user, [MacOSXHints, page 1, paragraph 1] ...and further configured to present the items without said corresponding metadata to the user in accordance with a

default display view, wherein said shell is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks” [MacOSXHints, page 2, paragraph 8 with MacOSXHints, pages 2-3, paragraphs 10 and 12].

MacOSXHints discloses the above limitations but does not expressly teach:

- “...wherein the shell is configured to present an item containing said metadata to the user with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata contained in said item.”

With respect to Claim 1, an analogous art, Landau, teaches:

- “...wherein the shell is configured to present an item containing said metadata to the user with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata contained in said item” [Landau, p. 4, everything below Fig. 6.43 with MacOSXHints, pages 1-2, paragraphs 3, 6, and 8].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to combine Landau with MacOSXHints because both inventions are directed towards storing and viewing files on file systems.

Landau’s invention would have been expected to successfully work well with MacOSXHints’s invention because both inventions use Mac OS X. MacOSXHints discloses Mac OS X’s handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties. However, MacOSXHints does not

Art Unit: 2161

expressly disclose that shell also presents the item that contains metadata to the user with the visual representation. Landau discloses TinkerTool and a manual operation comprising viewing invisible files.

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to take the viewing of invisible files from Landau and install it into the invention of MacOSXHints, thereby offering the obvious advantage of showing all system files to the user so that the user may be more informed on all the items the system uses and all the items that are present in a given area.

In this combination, since invisible files are made visible, a user can move them around like normal icons also in a window area. Since icon locations are saved, the locations of the .DS\_Store files would also be saved.

**Claim 2** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 1, wherein said universal data schema includes one or more declared properties” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 3** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 2, wherein each of said plurality of items includes declarations associated with said one or more declared properties” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 4** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 2, wherein said shell utilizes said one or more declared properties to present the items without said corresponding metadata in accordance with

said default display view” [MacOSXHints, page 2, paragraph 8 with MacOSXHints, pages 2-3, paragraphs 10 and 12].

**Claim 5** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 1, wherein said metadata includes item decoration data” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 6** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 5, wherein said item decoration data identifies a subset of said metadata containing high value data” [MacOSXHints, page 2, paragraphs 6 and 8 with MacOSXHints, pages 2-3, paragraphs 10-12].

**Claim 7** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 6, wherein said high value data is associated with information to be displayed to a user” [MacOSXHints, page 2, paragraphs 6 and 8 with MacOSXHints, pages 2-3, paragraphs 10-12].

**Claim 8** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 5, wherein said item decoration data includes data indicating a format aspect associated with presentment of said metadata to a user” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 9** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 8, wherein said format aspect is selected from the group consisting of data formatting, sort order, icon formatting, and associated controls” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 10** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 8, wherein said format aspect includes one or more verbs, said verbs being associated with operations to be performed on selected items”  
[MacOSXHints, page 2, paragraph 6 with MacOSXHints page 3, paragraphs 12-14].

**Claim 11** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 10, wherein said operations are performed by one or more applications launched to perform said verbs with respect to said selected items”  
[MacOSXHints, page 2, paragraphs 6 and 8 with MacOSXHints page 3, paragraphs 12-14].

**Claim 12** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 1, wherein said shell is further configured to present a set of items selected from said plurality of items according to a shell decoration view”  
[MacOSXHints, page 1, paragraph 1 with MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 13** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 12, wherein said set of items includes one or more items having metadata stored in accordance with a first item decoration schema and further includes one or more items having metadata stored in accordance with a second item decoration schema” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

**Claim 14** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 12, wherein said shell decoration view includes one or more display aspects appropriate for displaying each of said set of items to a user”  
[MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 15** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 14, wherein each of said set of items include a common characteristic” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 16** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 15, wherein said common characteristic is selected from the group consisting of document file, image file, audio file, and video file” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 17** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 12, wherein said one or more of said set of items are associated with an item decoration view that conflicts with said shell decoration view” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

**Claim 18** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The computer system of Claim 12, wherein said shell is further configured to present one or more of said set of items with decorative elements associated with an item decoration view and wherein said decorative elements do not conflict with said shell decoration view” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

For **Claim 30**, MacOSXHints teaches: “A computer implemented method for presenting a plurality of items stored in a universal data store to a user, [MacOSXHints, page 1, paragraph 3 with MacOSXHints, page 2, paragraphs 6 and 8] the method comprising:

- accessing said universal data store in response to a request to present one or more of said plurality of items to the user, [MacOSXHints, pages 2-3, paragraphs

Art Unit: 2161

6, 8, and 10-14] wherein said plurality of items are stored are accordance with a universal data schema, [MacOSXHints, page 1, paragraph 3 with MacOSXHints, page 2, paragraphs 6 and 8] and wherein at least a portion of said plurality of items contain metadata stored in accordance with an item decoration schema; [MacOSXHints, pages 1-2, paragraphs 3, 6, and 8] and

- ...presenting said one or more requested items to the user, [MacOSXHints, page 1, paragraph 1]...wherein a requested items without said corresponding metadata is presented in accordance with a default display view; and ” [MacOSXHints, page 2, paragraph 8 with MacOSXHints, pages 2-3, paragraphs 10 and 12]
- launching at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks” [MacOSXHints, pages 1-2, paragraphs 3, 6, and 8.]

MacOSXHints discloses the above limitations but does not expressly teach:

- “...for at least a portion of said one or more requested items, utilizing said metadata contained in a requested item to define a visual representation of said requested item and to define a plurality of user-selectable tasks;
- ...wherein at least a portion of said one or more requested items with said corresponding metadata are presented with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata.”

With respect to Claim 30, an analogous art, Landau, teaches:

- “...for at least a portion of said one or more requested items, utilizing said metadata contained in a requested item to define a visual representation of said requested item and to define a plurality of user-selectable tasks; [Landau, p. 4, everything below Fig. 6.43 with MacOSXHints, pages 1-2, paragraphs 3, 6, and 8]
- ...wherein at least a portion of said one or more requested items with said corresponding metadata are presented with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata” [Landau, p. 4, everything below Fig. 6.43 with MacOSXHints, pages 1-2, paragraphs 3, 6, and 8].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to combine Landau with MacOSXHints because both inventions are directed towards storing and viewing files on file systems.

Landau's invention would have been expected to successfully work well with MacOSXHints's invention because both inventions use Mac OS X. MacOSXHints discloses Mac OS X's handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties. However, MacOSXHints does not expressly disclose that what is presented is also the item that contains metadata to the user with the visual representation including one or more elements of visual content. Landau discloses TinkerTool and a manual operation comprising viewing invisible files.



It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to take the viewing of invisible files from Landau and install it into the invention of MacOSXHints, thereby offering the obvious advantage of showing all system files to the user so that the user may be more informed on all the items the system uses and all the items that are present in a given area.

In this combination, since invisible files are made visible, a user can move them around like normal icons also in a window area. Since icon locations are saved, the locations of the .DS\_Store files would also be saved.

**Claim 31** can be mapped to MacOSXHints (as modified by Landau) as follows:  
“The method of Claim 30, wherein said metadata includes data indicating one or more decorative aspects associated with presentment of said metadata to a user”  
[MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 32** can be mapped to MacOSXHints (as modified by Landau) as follows:  
“The method of Claim 31, wherein presenting a requested item with said corresponding metadata includes presenting said metadata according to one or more of said decorative aspects” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

**Claim 33** can be mapped to MacOSXHints (as modified by Landau) as follows:  
“The method of Claim 31, wherein said method further comprises presenting a set of items according to a shell decoration view, wherein said set of items includes one or more items having metadata stored in accordance with a first item decoration schema and further includes one or more items having metadata stored in accordance with a

second item decoration schema” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

**Claim 34** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The method of Claim 33, wherein said shell decoration view includes one or more display aspects appropriate for displaying each of said set of items to a user”

[MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 35** can be mapped to MacOSXHints (as modified by Landau) as follows:

“The method of Claim 33, wherein presenting said set of items according to a shell decoration view includes presenting items having item decoration schema which conflicts with said shell decoration view according to said shell decoration view”

[MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

For **Claim 41**, MacOSXHints teaches: “A shell embodied on one or more computer-readable media and executing on a computer for presenting a plurality of items stored in a universal data store to a user, [MacOSXHints, page 1, paragraph 3] the shell comprising:

- a data store interaction component which retrieves metadata contained within one or more of said plurality of items in response to a request to present the one or more items to the user, [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14] wherein said plurality of items are stored in accordance with a universal data schema, [MacOSXHints, page 1, paragraph 3 with MacOSXHints, page 2, paragraphs 6 and 8] and wherein at least a portion of said plurality of items contain said metadata defining a visual representation of at least one of said

plurality of items and defining a plurality of user-selectable tasks; [MacOSXHints, pages 1-2, paragraphs 3, 6, and 8] and

- a display presentation component which utilizes said retrieved metadata to present a display view of at least a portion of said one or more requested items, [MacOSXHints, page 1, paragraph 1] and further configured to present a requested item without said corresponding metadata in accordance with a default display view, wherein said display presentation component is further configured to launch at least one application in response to a user input selecting one or more of said plurality of user-selectable tasks” [MacOSXHints, pages 1-2, paragraphs 3, 6, and 8 with MacOSXHints, pages 2-3, paragraphs 10 and 12].

MacOSXHints discloses the above limitations but does not expressly teach:

- “wherein said display presentation component is configured to present a requested item containing said metadata to the user with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata contained in said requested item.”

With respect to Claim 41, an analogous art, Landau, teaches:

- “wherein said display presentation component is configured to present a requested item containing said metadata to the user with said visual representation and with an indication of said plurality of user-selectable tasks defined by said metadata contained in said requested item” [Landau, p. 4, everything below Fig. 6.43 with MacOSXHints, pages 1-2, paragraphs 3, 6, and 8].

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to combine Landau with MacOSXHints because both inventions are directed towards storing and viewing files on file systems.

Landau's invention would have been expected to successfully work well with MacOSXHints's invention because both inventions use Mac OS X. MacOSXHints discloses Mac OS X's handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties. However, MacOSXHints does not expressly disclose that what is presented is also the item that contains metadata to the user with the visual representation. Landau discloses TinkerTool and a manual operation comprising viewing invisible files.

It would have been obvious to one of ordinary skill in the art at the time of invention having the teachings of Landau and MacOSXHints before him/her to take the viewing of invisible files from Landau and install it into the invention of MacOSXHints, thereby offering the obvious advantage of showing all system files to the user so that the user may be more informed on all the items the system uses and all the items that are present in a given area.

In this combination, since invisible files are made visible, a user can move them around like normal icons also in a window area. Since icon locations are saved, the locations of the .DS\_Store files would also be saved.

13. Claims 19-23, 25-29, 36, 37, 39, and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacOSXHints

(<http://www.macoshints.com/article.php?story=20030305025744788>) in view of “Mac OS X Disaster Relief: Troubleshooting Techniques to Help Fix It Yourself” (Landau), further in view of U.S. Patent No. 5,937,406 (Balabine et al.).

For **Claim 19**, MacOSXHints (as modified by Landau) teaches: “The computer system of Claim 1.”

MacOSXHints (as modified by Landau) discloses the above limitations but does not expressly teach: “...wherein said shell is further configured to present items selected from said plurality of items according to an explorer display view.”

With respect to Claim 19, an analogous art, Balabine, teaches: “...wherein said shell is further configured to present items selected from said plurality of items according to an explorer display view” [Balabine, Figs. 5A-5C].

It would have been obvious to one of ordinary skill in the art at the time of invention to combine Balabine with MacOSXHints (as modified by Landau) because both inventions are directed towards storing and viewing files on file systems.

Balabine’s invention would have been expected to successfully work well with MacOSXHints (as modified by Landau)’s invention because both inventions use file systems on computers. MacOSXHints (as modified by Landau) discloses Mac OS X’s handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties, however MacOSXHints (as modified by Landau) does not

expressly disclose an explorer display view. Balabine discloses a file system interface to a database comprising an explorer display view.

It would have been obvious to one of ordinary skill in the art at the time of invention to take the explorer display view from Balabine and install it into the computer system of MacOSXHints (as modified by Landau), thereby offering the obvious advantage of offering the user a choice in display view interfaces (MacOSXHints, paragraph 11).

**Claim 20** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more decorative properties” [MacOSXHints, page 2, paragraph 6 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C].

**Claim 21** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more shell display views, and wherein one of said shell display views is utilized to display a set of explorer items” [MacOSXHints, page 2, paragraph 6 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C].

**Claim 22** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 21, wherein said shell display view includes one or more display aspects appropriate for displaying each of said set of explorer items to a user” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 23** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 22, wherein each of said set of

explorer items include a common characteristic” [MacOSXHints, page 2, paragraphs 6 and 8].

**Claim 25** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 21, wherein said explorer display view is configured to display an explorer item which is not associated with a shell display view according to an item decoration view” [MacOSXHints, page 2, paragraphs 6 and 8 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C].

**Claim 26** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 21, wherein said explorer display view is configured to display an explorer item which is not associated with an item decoration view according to said default display view” [MacOSXHints, page 2, paragraphs 6 and 8 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C].

**Claim 27** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more data queries associated with said explorer display view” [Balabine, col. 5, lines 25-30].

**Claim 28** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 19, wherein said explorer display view includes one or more verbs, said verbs being associated with an operation to be performed on a selected explorer item” [MacOSXHints, page 2, paragraph 6 with MacOSXHints page 3, paragraphs 12-14].

**Claim 29** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The computer system of Claim 28, wherein said operation is performed by an application launched to perform at least one of said one or more verbs with respect to said selected explorer item” [with MacOSXHints, pages 1-2, paragraphs 3, 6, and 8 with MacOSXHints page 3, paragraphs 12-14].

For **Claim 36**, MacOSXHints (as modified by Landau) teaches: “The method of Claim 30.”

MacOSXHints (as modified by Landau) discloses the above limitations but does not expressly teach: “...wherein said method further comprises presenting items according to an explorer display view.”

With respect to Claim 36, an analogous art, Balabine, teaches: “...wherein said method further comprises presenting items according to an explorer display view” [Balabine, Figs. 5A-5C].

It would have been obvious to one of ordinary skill in the art at the time of invention to combine Balabine with MacOSXHints (as modified by Landau) because both inventions are directed towards storing and viewing files on file systems.

Balabine's invention would have been expected to successfully work well with MacOSXHints (as modified by Landau)'s invention because both inventions use file systems on computers. MacOSXHints (as modified by Landau) discloses Mac OS X's handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties, however MacOSXHints (as modified by Landau) does not



expressly disclose an explorer display view. Balabine discloses a file system interface to a database comprising an explorer display view.

It would have been obvious to one of ordinary skill in the art at the time of invention to take the explorer display view from Balabine and install it into the computer system of MacOSXHints (as modified by Landau), thereby offering the obvious advantage of offering the user a choice in display view interfaces (MacOSXHints, paragraph 11).

**Claim 37** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The method of Claim 36, wherein said explorer display view includes one or more decorative properties [MacOSXHints, page 2, paragraph 6 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C] and one or more shell display views, and wherein one or more of said shell display views are utilized to display a set of explorer items” [MacOSXHints, page 2, paragraph 6 with MacOSXHints, page 3, paragraphs 11-13 with Balabine, Figs. 5A-5C].

**Claim 39** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The method of Claim 37, wherein an item which cannot be displayed according to a shell display view is presented according to an item decoration view” [MacOSXHints, pages 2-3, paragraphs 6, 8, and 10-14].

**Claim 40** can be mapped to MacOSXHints (as modified by Landau and Balabine) as follows: “The method of Claim 36, wherein said explorer display view includes one or more verbs, said verbs being associated with an operation to be

Art Unit: 2161

performed on a selected explorer item” [MacOSXHints, page 2, paragraph 6 with MacOSXHints page 3, paragraphs 12-14].

14. Claims 24 and 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacOSXHints

(<http://www.macosxhints.com/article.php?story=20030305025744788>) in view of “Mac OS X Disaster Relief: Troubleshooting Techniques to Help Fix It Yourself” (Landau) in view of U.S. Patent No. 5,937,406 (Balabine et al.), further in view of MacOSXHints2 (<http://www.macosxhints.com/article.php?story=20030409015020645>).

For **Claim 24**, MacOSXHints (as modified by Landau and Balabine) teaches: “The computer system of Claim 21.”

MacOSXHints (as modified by Landau and Balabine) discloses the above limitations but does not expressly teach: “...wherein said explorer display view is configured to utilize a shell display view not included with an explorer to display an item which cannot be displayed according to said one or more shell display views included with said explorer display view.”

With respect to Claim 24, an analogous art, MacOSXHints2, teaches: “...wherein said explorer display view is configured to utilize a shell display view not included with an explorer to display an item which cannot be displayed according to said one or more shell display views included with said explorer display view” [MacOSXHints2, page 1 with MacOSXHints, page 1, paragraph 3].

It would have been obvious to one of ordinary skill in the art at the time of invention to combine MacOSXHints2 with MacOSXHints (as modified by Landau and Balabine) because both inventions are directed towards storing and viewing files on file systems.

MacOSXHints2's invention would have been expected to successfully work well with MacOSXHints (as modified by Landau and Balabine)'s invention because both inventions use file systems on computers. MacOSXHints (as modified by Landau and Balabine) discloses Mac OS X's handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties, however MacOSXHints (as modified by Landau and Balabine) does not expressly disclose an explorer display view. MacOSXHints2 discloses changing the finder preference of viewing invisible files comprising a script.

It would have been obvious to one of ordinary skill in the art at the time of invention to take the explorer display view from MacOSXHints2 and install it into the computer system of MacOSXHints (as modified by Landau and Balabine), thereby offering the obvious advantage of offering the user a choice in display view interfaces (MacOSXHints, paragraph 11).

For **Claim 38**, MacOSXHints (as modified by Landau and Balabine) teaches:  
"The method of Claim 37."

MacOSXHints (as modified by Landau and Balabine) discloses the above limitations but does not expressly teach: "...wherein an item which cannot be displayed according to said one or more shell display views included with said explorer display

Art Unit: 2161

view is presented according to a shell display view not included with said explorer display view.”

With respect to Claim 38, an analogous art, MacOSXHints2, teaches: “...wherein an item which cannot be displayed according to said one or more shell display views included with said explorer display view is presented according to a shell display view not included with said explorer display view” [MacOSXHints2, page 1 with MacOSXHints, page 1, paragraph 3].

It would have been obvious to one of ordinary skill in the art at the time of invention to combine MacOSXHints2 with MacOSXHints (as modified by Landau and Balabine) because both inventions are directed towards storing and viewing files on file systems.

MacOSXHints2's invention would have been expected to successfully work well with MacOSXHints (as modified by Landau and Balabine)'s invention because both inventions use file systems on computers. MacOSXHints (as modified by Landau and Balabine) discloses Mac OS X's handling of metadata display information comprising a .DS\_Store file, windows, icons, and display properties, however MacOSXHints (as modified by Landau and Balabine) does not expressly disclose an explorer display view. MacOSXHints2 discloses changing the finder preference of viewing invisible files comprising a script.

It would have been obvious to one of ordinary skill in the art at the time of invention to take the explorer display view from MacOSXHints2 and install it into the computer system of MacOSXHints (as modified by Landau and Balabine), thereby

offering the obvious advantage of offering the user a choice in display view interfaces (MacOSXHints, paragraph 11).

15. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

### **Conclusion**

16. Any prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is advised that, although not used in the rejections above, prior art cited on any PTO-892 form and not relied upon is considered materially relevant to the applicant's claimed invention and/or portions of the claimed invention.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Brent S. Stace whose telephone number is 571-272-8372 and fax number is 571-273-8372. The examiner can normally be reached on M-F 9am-5:30pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Apu M. Mofiz can be reached on 571-272-4080. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/B. S./  
Examiner, Art Unit 2161

/Apu M Mofiz/  
Supervisory Patent Examiner, Art Unit 2161